REMARKS

The above amendment and these remarks are responsive to the Office Action by Examiner Pierre M. Bataille, mailed 17 July 2003, which reopened prosecution in view of appellant's brief filed on 12 August 2002.

Withdrawn Claims 2-30

The Examiner characterizes claims 2-30 as withdrawn. These claims were subject to restriction in the parent application of this divisional case, and were prosecuted in the parent or two other divisional cases. Original claim 1 is being prosecuted in this divisional case, along with claims 31-39 which were previously added.

Claims 1 and 31-39 are in the case, none having been allowed.

35 U.S.C. 102

Claims 1 and 31-39 have been rejected under 35 U.S.C.

102(e) as being anticipated by Steely, Jr., et al. U.S. Patent 6,249,520 (hereinafter, Steely, Jr.).

Applicants traverse, and argue that the Examiner has not made the required prima facie case of anticipation, which requires that the Examiner provides

1. a single reference
2. that teaches or enables
3. each of the claimed elements (arranged as in the claim)
4. expressly or inherently
5. as interpreted by one of ordinary skill in the art.

With respect to the third element, it is not enough that the prior art reference disclose all the claimed elements in isolation. Rather, as stated by the Federal Circuit, the prior art reference must disclose each element of the claimed invention "arranged as in the claim." The Examiner, further, must identify the elements of the claims of the application, determine their meaning in light of the specification and prosecution history, and identify the corresponding elements disclosed in the allegedly

anticipating reference. Further, anticipation will not be found when the prior art is lacking or missing a specific feature or structure of the claimed invention.

The Examiner correctly states that Steely, Jr. teaches certain aspects of the present invention, such as a shared memory parallel processing system having a cache coherency system including a plurality of nodes interconnected through an interconnecting network; where each processing node includes a unique section of shared memory, caches for storing a plurality of cache lines, and a distributed cache directory for tracking which of said nodes have a copy of each cache line.

Cache Coherency System

However, the cache coherency system of applicant is significantly different from that of Steely, Jr. which teaches a switch and uses a shared memory system implementation as a backdrop to show how his switch invention works.

In contrast, the present invention teaches a cache coherency system and uses a shared memory system

implementation as a backdrop to show how the cache coherency system works. The fact that both inventions use a shared memory system as a backdrop for teaching the respective inventions in no way implies that the inventions are the same or that one could be anticipated from the other.

Although the backdrops for teaching the inventions are similar, the present invention teaches a significantly different cache coherency system than Steely, Jr., as will be further substantiated below. Since the present invention is a cache coherency system, theoretically, only the cache coherency system needs to be proven different than Steely, Jr. to show the uniqueness of the solution. However, it will also be shown that there are other differences between the two systems.

"Ownership Protocol" vs. "Write-Thru Protocol"

Applicant respectfully disagrees with the Examiner that Steely, Jr. teaches a cache coherency system wherein "an adapter for storing changed data immediately to said unique section of shared memory regardless of which said node is changing the data and which of said nodes includes the section of shared memory to be changed, wherein said

shared memory always contains the most recent data."

Steel, Jr. teaches an entirely different cache coherent protocol described in detail starting at Col. 31 and entitled "Cache Coherency Protocol". The protocol Steel, Jr. teaches is an "ownership protocol" because there is always an identifiable "owner" of each cache line. The "owner" can be memory, one of the processors or one of the IOP's in the system (Col. 31, line 7-9). The "owner" of each cache line is identified by the contents (a 14 bit word for each cache line) of DTAG 20 of Figure 2. This is explained in detail in the DTAG and IOP Tag section of the text starting on Col. 16, line 42. The DTAG "directory" is used to provide "ownership" information for each block of memory in the DTAG of the associated multiprocessing node (the home node) (Col. 17, lines 38-40). The DTAG directory stores one 14 bit word for each 64 byte block of data. In Fig. 9 the 14 bit DTAG entry word "is shown to include an owner ID field 142" (Col. 17, lines 57-58) that identifies the owner of a particular 64 byte block of memory. The owner ID field comprises 6 bits of the 14 bits (Col. 17, line 65-66) and defines the current "owner" of the block as either one of 32 processors, one of eight IOP processors or memory" (Col.18, lines 1-3).

Thus, Steely, Jr. teaches assignable ownership for each cache line, and that modifying (storing to) a cache line "owned" by a processor on another node of the network is a very complicated process as described starting at Col. 35, line 1. The process of storing modified data includes all sorts of complexities such as network generated probe commands (including forwards, markers and fills) plus invalidates, clean to dirty transactions, exclusive ownership, and victim transfers.

Steely, Jr's system of cache coherency is very complex and totally different from the present invention which teaches and variously claims an adapter for storing changed data immediately to an unique section of shared memory regardless of which node is changing the data and which of the nodes includes the section of shared memory to be changed, wherein the shared memory always contains the most recent data.

The present invention uses none of the complexities taught by Steely Jr. On this point alone, Steely Jr. fails to teach the present claimed invention.

Network Affect on Cache Coherency

One of the reasons that Steely, Jr. does not teach an immediate storage of changed memory data to shared memory is that Steely, Jr. have a much inferior switch and network that is incapable of immediately storing data to nodes across the switch network. This is because Steely, Jr's network is a very slow network containing congestion, blocking, buffers in the network, priority schemes and complex functioning. To add even further complexity to Steely, Jr's switch network it is divided into multiple sub-networks where each 4-processor node contains a first network as shown in Figure 2 comprised of network 19 and network control chip 18 (shown in more detail in Figure 2). Steely, Jr's second network called the "hierarchical switch 155" in Figure 7A interconnects eight first networks 19, since each block 100 of figure 7A incorporates one first network 19. Figure 15 shows the complexity of the switch buffers internal to switch network 155, where any data traversing the switch can get bogged down. Figure 16 shows some of the control for handling the complexity. Steely, Jr. describes this complexity starting at Col. 21, line 36 and going to Col. 30, line 66.

In contrast, Figure 2A of the present invention shows that each node 30

> "...interfaces to network 20 via a network adapter 10. Node 30 includes processor 50, system memory 54, and I/O controller 52, and network adapter 10. Node 30 attaches to one port 23A of the network 20 in full duplex and contains network adapter 10 which sends to and receives messages from the network 20 for communication with other nodes 34 (pg. 24, lines 1-8)."

Thus, each node of the present system has only one processor which connects directly to the network switch 60X of Figure 5. The present invention has only a single network to which each processor is directly connected, and doesn't split the network into first and second networks as Steely, Jr. does. This network organization is one factor which allows the present system to store data immediately to shared memory, even at other nodes. The other network factor is the speed of the switch network in the present invention provided by bufferless switches, alternate paths thru the network (pg. 29, lines 22-24) to get around blockages, and two modes of switch operation for improved efficiency.

The preferred embodiment of applicant's network 20 is a multi-stage interconnection network comprised of Allnode switches at each stage of network 20 of Figure 4. The dual priority version of the Allnode switch (US Patent 5,444,705, "Dual Priority Switching Apparatus for Simplex Networks") provides the switch which has multiple copies interconnected to form network 20 for this invention. The Allnode dual priority switch is called dual because it operates in two basic modes: 1) normal or low priority mode, and 2) camp-on or high priority mode. (Pg. 28 lines 13-22). The switch is also easily expandable to systems with more nodes because of the simplicity of the network (Pg. 31, lines 20-22). Thus, the speed and efficiency of the network in the present invention permits the unique concept of storing changed data immediately to shared memory even over the network.

Network Hops Required for Reading Shared Memory Data

Because of the slower network, Steely, Jr. uses an "ownership protocol". The "ownership protocol is very complex and even requires a special ARB bus (Col. 31, lines 43-50), special processor instructions which involve the processor in controlling the coherency protocol including Change-to-Dirty Commands, Victim Commands, and Evict

commands (Col. 32, lines 23-30), and special hardware to control the process by issuing Probe and Response commands (Col. 32, lines 31- 65).

Like the "write back" cache approach discussed previously in responses to Office Actions regarding this case, Steely, Jr's "ownership protocol" also uses a 3 or more hop method (Col. 35, line 23) to acquire data over the network. A "requesting node" must query a "home node" for the location of which node and cache contains the ownership of the most recent data (hop 1). The "home node" searches its D-Tag to determine the ownership of the processor cache holding the requested data. Then, a message is sent from the home node to the ownership node requesting that a copy of the cache line be sent to the requesting node (hop 2). The ownership node sends the requested data to the requesting node (hop 3). In addition, the home node sends a "Fill Marker" message back to the requesting node, which acts as a fourth network transmission, further delaying and clogging the network. Steely, Jr. describes this process at Col. 33, line 57 to Col. 34, line 21 as one of the many complex transfers that can occur.

The present invention does not teach or require a

three-hop or four-hop method, where the cache of any processor node can store the most recent data. In contrast, the present application teaches a "write-thru" cache coherency method, whereby any node changing the data immediately writes the most recent data back to the shared memory at the home node, such that the shared memory at home node always has the most recent data, and the home node never has to search any other nodes or caches for where the most recent data is stored.

The present invention implements a two-hop method as follows: hop 1) a requesting node requests the most recent data of the home node, and hop 2) the home immediately returns the most recent data from its shared memory to the requesting node. This provides the quickest response possible for fetching global data - there can be nothing faster or simpler than the method taught by the present application. It also greatly reduces network traffic, and is a far superior solution. The present invention never involves the processor in cache coherency decisions nor is the processor required to issue special commands to aid the hardware. The network adapter 10 and memory controller 210 of the present invention shown in Figures 11 and 15, respectively, provide a hardwired implementation for

controlling both network transfers and cache coherency without burdening the processor in any way. Therefore, it is unique and a vast improvement over Steely, Jr. et al. The present invention does not require "ownership protocol", operation complexity, excessive hops, processor involvement, Probe and Response commands, or Arb Bus implementation and control.

First and Second Memory Portions

On page 4 of the Office Action, the Examiner asserts that Steely, Jr. discloses a shared memory including a first memory portion for storing unchangeable data and a second memory portion for storing changeable data.

Applicant respectfully disagrees.

Nowhere does Steely, Jr. teach this concept. The Examiner sites Col. 14, Lines 22-30. These lines describe the terms for local memory vs. remote memory, which indicates whether the memory is located at the same node in which the processor resides or at a node across the network. This text does not teach memory sections dedicated to unchangeable or changeable data. The Examiner also sites

Col. 18, lines 18-64. These lines describe the operation of the DTAG and have no relevance to memory sections dedicated to unchangeable or changeable data.

The claims of the present invention require no further modification as they already clearly differentiate unique aspects over Steely, Jr., including recitation of write-thru caches storing the most recent data directly to shared memory over a multi-stage network. Therefore, Applicant requests that claims 1, and 31-39 be allowed as they stand.

SUMMARY AND CONCLUSION

Applicant urges that the case be passed to issue with claims 1 and 31-39.

If, in the opinion of the Examiner, a telephone conversation with applicant(s) attorney could possibly facilitate prosecution of the case, he may be reached at the number noted below.

Sincerely,

Howard T. Olnowich

By

Shelley M Beckstrand
Reg. No. 24,886

Date: 18 August 2003

Shelley M Beckstrand, P.C.
Attorney at Law
314 Main Street
Owego, NY 13827

Phone: (607) 687-9913
Fax: (607) 687-7848